eSi-SPI Flash

1 Contents

1 Contents 2
2 Overview 3
3 Hardware Interface 4
4 Software Interface 6
4.1 Register Map 6
4.2 Interrupts 10
4.3 SPI Command Codes 11
5 Revision History 12

2 Overview

The eSi-SPI Flash core can be used to provide both a memory mapped interface and control register based interface to a serial flash memory device. It supports the following features:

- Execute-in-place (XIP) support to allow code to execute directly from flash.
- Programmable bit rate.
- Support for serial (1-1-1), dual (1-1/2-2) and quad (1-1/4-4) mode SPI flash devices.
- Support for auto-incrementing addresses (saves transmitting sequential addresses).
- Support for I/O mode (transmits address in dual/quad mode as well as data).
- Support for continuous mode (saves transmitting read commands).
- Support for 24-bit and 32-bit addresses.
- Support for single data rate (SDR) and double data rate (DDR) transfers.
- Pins can be reused as GPIOs when flash not in use.
- AMBA 3 AHB-lite memory mapped slave interface.
- AMBA 3 AHB-lite control slave interface.

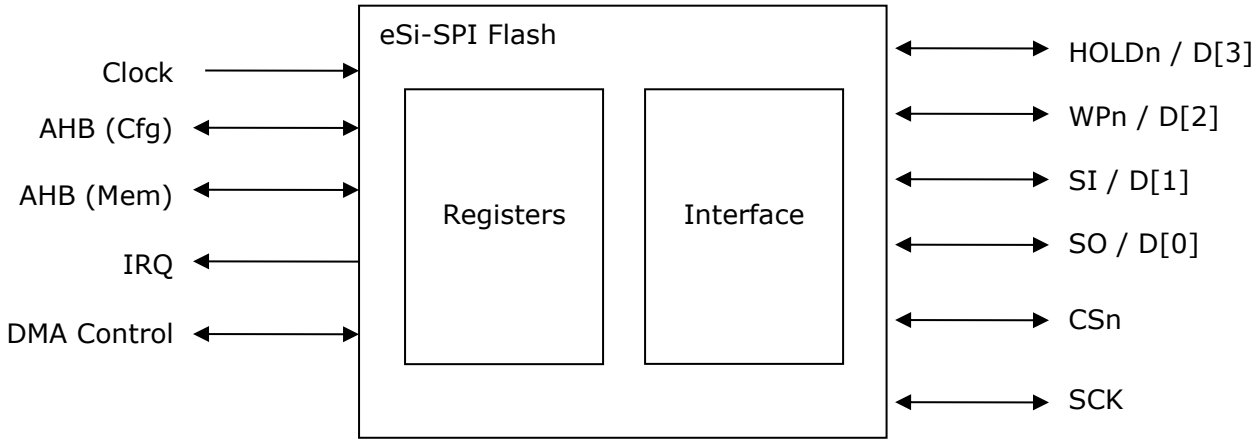

Figure 1: eSi-SPI Flash

3 Hardware Interface

Module Name	cpu_ahb_spi_flash
HDL	Verilog
Technology	Generic
Source Files	cpu_ahb_spi_flash.v, cpu_peripheral_flow_control.v

Port	Type	Description
address_width	Integer	Specifies how many of the least significant bits of mem_haddr are used to form the address sent to flash
cycles_per_bit_reset	Integer	Specifies the reset value of the cycles_per_bit register
reset_delay_cycles	Integer	Specifies how many AHB clock cycles to wait after reset before chip-select can be asserted
reset_address	Integer	Specifies the reset value for the address register
reset_address_size	Integer	Specifies the reset value for control.AS
reset_read_dummy_cycles	Integer	Specifies the reset value for timing.RD

Table 1: Parameters

Port	Direction	Width	Description
mem_hclk	Input	1	Memory interface, AHB clock
mem_hresetn	Input	1	Memory interface, AHB reset, active-low
mem_haddr	Input	BITS	Memory interface, AHB address
mem_hburst	Input	3	Memory interface, AHB burst type
mem_hmastlock	Input	1	Memory interface, AHB locked transfer
mem_hprot	Input	4	Memory interface, AHB protection
mem_hsize	Input	3	Memory interface, AHB size
mem_htrans	Input	2	Memory interface, AHB transfer type
mem_hwdata	Input	BITS	Memory interface, AHB write data
mem_hwrite	Input	1	Memory interface, AHB write
mem_hready	Input	1	Memory interface, AHB ready
mem_hsel	Input	1	Memory interface, AHB select
mem_hready	Output	1	Memory interface, AHB ready
mem_hrdata	Output	BITS	Memory interface, AHB read data
mem_hresp	Output	1	Memory interface, AHB response
cfg_hclk	Input	1	Configuration interface, AHB clock
cfg_hresetn	Input	1	Configuration interface, AHB reset, active-low
cfg_haddr	Input	BITS	Configuration interface, AHB address
cfg_hburst	Input	3	Configuration interface, AHB burst type
cfg_hmastlock	Input	1	Configuration interface, AHB locked transfer
cfg_hprot	Input	4	Configuration interface, AHB protection
cfg_hsize	Input	3	Configuration interface, AHB size
cfg_htrans	Input	2	Configuration interface, AHB transfer type
cfg_hwdata	Input	BITS	Configuration interface, AHB write data
cfg_hwrite	Input	1	Configuration interface, AHB write
cfg_hready	Input	1	Configuration interface, AHB ready
cfg_hsel	Input	1	Configuration interface, AHB select
cfg_hready	Output	1	Configuration interface, AHB ready
cfg_hrdata	Output	BITS	Configuration interface, AHB read data
cfg_hresp	Output	1	Configuration interface, AHB response
cs_n_in	Input	1	Chip-select input (for use as GPIO)
sck_in	Input	1	Serial clock input (for use as GPIO)
si_in	Input	1	Serial data input
so_in	Input	1	Serial data output, input value

wp_n_in	Input	1	Write protect input value
hold_n_in	Input	1	Hold input value
tx_ack	Input	1	Acknowledges tx_ready after transfer complete
rx_ack	Input	1	Acknowledges rx_ready after transfer complete
mem_hclk_cactive	Output	1	Indicates mem_hclk should be active
cs_n	Output	1	Chip-select, active-low
cs_n_out_enable	Output	1	Chip-select output enable
sck	Output	1	Serial clock
sck_out_enable	Output	1	Serial clock output enable
si_out	Output	1	Serial data input, output value / quad data bit 1
si_out_enable	Output	1	Serial data input pin, output enable
so_out	Output	1	Serial data output / quad data bit 0
so_out_enable	Output	1	Serial data output enable
wp_n_out	Output	1	Write protect, active-low / quad data bit 2
wp_n_out_enable	Output	1	Write protect output enable
hold_n_out	Output	1	Hold, active low / quad data bit 3
hold_n_out_enable	Output	1	Hold output enable
interrupt_n	Output	1	Interrupt request, active-low
tx_ready	Output	1	Indicates device can accept new data
rx_ready	Output	1	Indicates device has data to be read

Table 2: I/O Ports

For complete details of the AHB signals, please refer to the AMBA 3 AHB-Lite Protocol v1.0 Specification available at:

http://www.arm.com/products/system-ip/amba/amba-open-specifications.php

4 Software Interface

4.1 Register Map

Register	Address offset	Access	Description
tx_data	0x00	W	Transmit register
rx_data	0x04	R	Receive register
status	0x08	R/W	Status register
control	0x0c	R/W	Control register
cycles_per_bit	0x10	R/W	Cycles per bit register
pin_out	0x14	R/W	Pin output data register
pin_in	0x18	R	Pin input data register
pin_direction	0x1c	R/W	Pin direction register
timing	0x20	R/W	Timing register
address	0x24	R/W	Address register

Table 3: Register Map

4.1.1 Transmit Data Register

Data to be transmitted over the serial interface should be written to the transmit register. The transmit data register should not be written to while the TXF bit in the status register is set, otherwise data loss may occur.

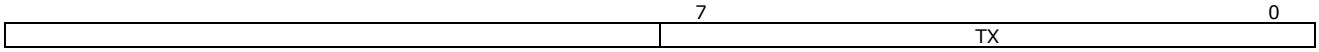

Figure 2: Format of the tx_data register

4.1.2 Receive Data Register

Data that is received over the serial interface can be read in the receive register.

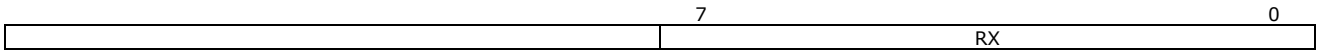

Figure 3: Format of the rx_data register

4.1.3 Status Register

The status register contains a selection of flags that indicate the current status of the SPI flash interface. To clear a bit in the status register, write a 1 to it. Writing 0 will leave it unchanged.

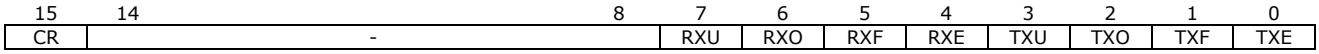

Figure 4: Format of the status register

Register	Values	Description
TXE	0 - Not empty 1 - Empty	Transmit buffer empty
TXF	0 - Not full 1 - Full	Transmit buffer full
TXO	0 - No overflow	Transmit buffer overflow

	1 - Overflow	
TXU	0 - No underrun 1 - Underrun	Transmit buffer underrun
RXE	0 - Not empty 1 - Empty	Receive buffer empty
RXF	0 - Not full 1 - Full	Receive buffer full
RXO	0 - No overflow 1 - Overflow	Receive buffer overflow
RXU	0 - No underrun 1 - Underrun	Receive buffer underrun
CR	0 – Not ready 1 – Ready	Indicates whether the interface is ready to accept transactions after switching to config mode. This must be checked to ensure any memory mode transactions are complete

Table 4: Fields of the `status` register

4.1.4 Control Register

The control register contains a selection of flags that control the operation of the SPI Flash interface.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
M	CE	AI	CR	CM	AM	DM		RXIE	TXIE	DDR	ODD	EDG	-	AS	E

Figure 5: Format of the `control` register

Register	Values	Description
E	0 - Disabled 1 - Enabled	Enables the SPI flash interface. When disabled, data will not be received or transmitted.
AS	0 – 3 bytes (24-bit) 1 – 4 bytes (32-bit)	Size of address to transmit to SPI Flash.
EDG	0 – Rising edge 1 – Falling edge	Determines which edge of `sck_out` receive data is sampled on
ODD	0 – Even 1 – Odd	Determines whether the `sck_out` clock divider denominator is even or odd. See the description for the `cycles_per_bit` register.
DDR	0 – Single data rate 1 – Double data rate	In single data rate mode the address and data rate are driven on the falling clock edge and sampled on the rising clock edge. In double data rate mode, address and data are driven and sampled on both edges of the clock.
TXIE	0 - Disabled 1 - Enabled	Transmit interrupt enable.
RXIE	0 - Disabled 1 - Enabled	Receive interrupt enable.
DM	0 – Serial mode 1 – Dual mode 2 – Quad mode	Serial, dual or quad mode select. In serial mode, SI pin is used for read data. In dual mode, SI & SO are used as a 2-bit parallel data bus. In quad mode, SI, SO, WPn and HOLDn are used as a 4-bit parallel data bus. Only serial mode is supported in config mode.
AM	0 – Output mode 1 – I/O mode	In output mode the address is transmitted serially, and the parallel bus is used only for data. In I/O mode, the parallel bus is used for

		both address and data.
CM	0 – Normal mode 1 – Continuous mode	In normal mode, the read command is transmitted to the SPI flash for every non-sequential access. In continuous mode, the read command is only transmitted once.
CR	0 – No reset 1 – Continuous reset	When set to 1, a continuous mode exit command will be transmitted, to remove the device from continuous mode. This bit will be cleared automatically after the reset is transmitted.
AI	0 – No auto-increment 1 – Auto-increment	When set to 1, it is assumed that the read address will automatically be incremented for each byte read while chip-select is held asserted. When set to 0, the chip-select will be deasserted after each byte and a new read command and address will be transmitted.
CE	0 – Stall 1 – Error	Determines whether accesses via the memory interface stall (mem_hready=0) or raise an error (mem_hresp=1), when in configuration mode (control.M=1).
M	0 – Memory mode 1 – Config mode	Determines whether the interface operates as a memory (i.e. it will respond to transactions on the mem_ AHB interface) or in configuration mode (i.e. SPI transactions can be performed via the tx_data and rx_data registers, or the pin registers and be used).

Table 5: Fields of the control register

4.1.5 Cycles Per Bit Register

The cycles per bit register is a 16-bit that specifies how many cycles of the clock, mem_hclk, it takes to transmit a bit. Use of a 16-bit register provides support for a wide range of clock frequencies and bit rates.

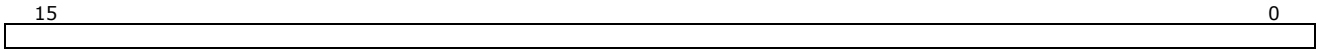

Figure 6: Format of the cycles_per_bit register

The frequency of sck_out is:

```
mem_hclk_freq / ((2 * (cycles_per_bit + 1)) + control.ODD)
```

When control.ODD is 0, the duty cycle of sck_out will be 50:50. When control.ODD is 1, the duty cycle will be biased low.

4.1.6 Pin Out Register

The pin out register holds the value that is to be output on the SPI flash interface pins, when a transfer is not taking place that uses them.

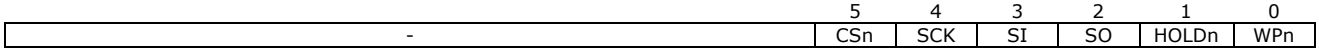

Figure 7: Format of the pin_out register

Register	Values	Description
WPn	0 – Output 0 1 – Output 1	Value to output on WPn pin
HOLDn	0 – Output 0 1 – Output 1	Value to output on HOLDn pin
SO	0 – Output 0 1 – Output 1	Value to output on SO pin
SI	0 – Output 0 1 – Output 1	Value to output on SI pin
SCK	0 – Output 0 1 – Output 1	Value to output on SCK pin
CSn	0 – Output 0 1 – Output 1	Value to output on CSn pin

Table 6: Fields of the `pin_out` register

4.1.7 Pin In Register

The pin in register holds the current value on the SPI flash interface pins.

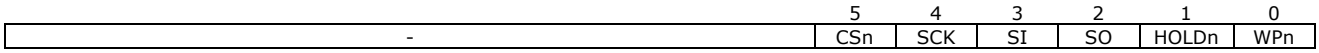

Figure 8: Format of the `pin_in` register

Register	Values	Description
WPn		Value on WPn pin
HOLDn		Value on HOLDn pin
SO		Value on SO pin
SI		Value on SI pin
SCK		Value on SCK pin
CSn		Value on CSn pin

Table 7: Fields of the `pin_in` register

4.1.8 Pin Direction Register

The pin direction register holds a direction flag for each bi-directional SPI flash interface pin, controlling whether it is an input or output.

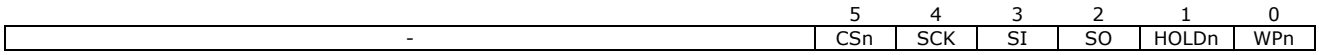

Figure 9: Format of the `pin_direction` register

Register	Values	Description
WPn	0 – Input 1 – Output	Direction of WPn pin
HOLDn	0 – Input 1 – Output	Direction of HOLDn pin
SO	0 – Input 1 – Output	Direction of SO pin
SI	0 – Input 1 – Output	Direction of SI pin
SCK	0 – Input 1 – Output	Direction of SCK pin

CSn	0 – Input 1 – Output	Direction of CSn pin

Table 8: Fields of the `pin_direction` register

4.1.9 Timing Register

The timing register holds fields to control the timing of the SPI interface.

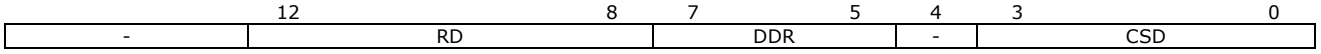

Figure 10: Format of the `timing` register

Register	Values	Description
CSD	0 - 15	Chip-select deassert AHB clock cycles. This is the minimum number of AHB clock cycles the chip-select pin is de-asserted for between accesses
DDR	0 – 7	When `control.DDR` = 1, this is the number of AHB clock cycles after an `sck_out` edge that the data outputs are driven
RD	0 - 31	Read dummy SPI clock cycles. This is the number of dummy SPI clock cycles required between a read command and when the data from the SPI flash will be valid. When `control.DDR` = 1, this should be 2x the number of SCK dummy cycles.

Table 9: Fields of the `timing` register

4.1.10 Address Register

The address register holds the next sequential 32-bit address to be accessed in the flash. The lower order bits of this register are set to `mem_addr[address_width-1:0]` each time the flash is accessed by the memory AHB interface. As `address_width` may be less than 32, this register allows the most significant bits of the flash address to be set.

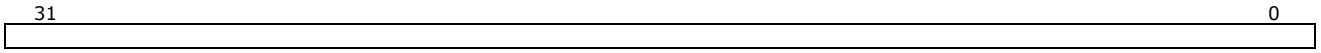

Figure 11: Format of the `address` register

4.2 Interrupts

The SPI flash interface supports the following interrupts.

- Transmit interrupt
- Receive interrupt

The transmit interrupt will be raised when the transmit buffer is empty and the `TXIE` flag in the `control` register is set to 1. This indicates that the transmitter has no data to transmit.

The receive interrupt will be raised when the receiver buffer is not empty and the `RXIE` flag in the control register is set to 1. This indicates that the receiver has received some data.

4.3 SPI Command Codes

The eSi-SPI Flash issues the command codes in Table 10.

Command	Code	control.DM	control.AS	control.DDR
Serial Read	h03	0	0	0
Dual Read	h3b	1	0	0
Quad Read	h6b	2	0	0
Dual IO Read	hbb	1	0	0
Quad IO Read	heb	2	0	0
4-byte Serial Read	h13	0	1	0
4-byte Dual Read	h3c	1	1	0
4-byte Quad Read	h6c	2	1	0
4-byte Dual IO Read	hbc	1	1	0
4-byte Quad IO Read	hec	2	1	0
DDR Dual IO Read	hbd	1	0	1
DDR Quad IO Read	hed	2	0	1
4-byte DDR Dual IO Read	hbe	1	1	1
4-byte DDR Quad IO Read	hee	2	1	1
Continuous mode reset	hff	`control.CR`=1		

Table 10: Command Codes

5 Revision History

Hardware Revision	Software Release	Description
1	2.7.0	Initial release
3	3.3.16	Added `address` register Added `cs_n_out_enable` and `sck_out_enable` ports Added `pin_direction.SCK` and `pin_direction.CSn` register fields
4	4.1.1	Added `control.CE` field
5	4.1.15	Add support for continuous mode via `control.CM`
6	5.0.7	Added `control.AS` field Added `control.EDG` field Added `control.ODD` field Added `control.DDR` field Added `timing.DDR` field Added `reset_address_size` parameter Added `reset_read_dummy_cycles` parameter Increased `address` register to 32 bits

Table 11: Revision History

